



**UNifeob**  
| ESCOLA DE NEGÓCIOS



2023

# PROJETO INTEGRADO



UNIFEOB  
CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS  
ESCOLA DE NEGÓCIOS  
**A.D.S. E CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO/  
IOT DATA STREAMER  
SIMPLE IOT**

SÃO JOÃO DA BOA VISTA, SP

NOVEMBRO 2023

UNIFEOB  
CENTRO UNIVERSITÁRIO DA FUNDAÇÃO DE ENSINO  
OCTÁVIO BASTOS  
ESCOLA DE NEGÓCIOS  
**A.D.S. E CIÊNCIA DA COMPUTAÇÃO**

**PROJETO INTEGRADO**

**IOT DATA STREAMER**

**SIMPLE IOT**

MÓDULO MODELAGEM E DESENVOLVIMENTO DE SISTEMAS

Gestão Financeira – Profa. Renata Elizabeth de Alencar Marcondes

Programação Orientada a Objeto – Prof. Nivaldo Andrade

Lógica de Programação – Prof. Marcelo Ciacco de Almeida

Modelagem de Dados – Prof. Max Streicher Vallim

Projeto de Modelagem e Desenvolvimento de Sistemas – Profª. Mariângela

Martimbianco Santos

Estudantes:

Caic Sant'Anna, RA 23000417

Everton Henrique Lopes, RA 23000201

Fábio Koiti Konda, RA 23000897

Gesiel dos Santos Guirra, RA 23000644

Lucas Pizol Ferreira, RA 23000019

Vitor Alexandre Rocetti Rinke, RA 23000081

# SUMÁRIO

1. INTRODUÇÃO	4
2. DESCRIÇÃO DA EMPRESA	5
3. PROJETO INTEGRADO	6
3.1 PROGRAMAÇÃO ORIENTADA A OBJETO	6
3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.	7
3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS	8
3.2 LÓGICA DE PROGRAMAÇÃO	8
3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE	9
3.2.2 DESENVOLVIMENTO DE APLICAÇÕES DESKTOP	10
3.3 MODELAGEM DE DADOS	10
3.3.2 MODELO LÓGICO E FÍSICO	10
3.3.3 SQL	11
3.4 GESTÃO FINANCEIRA	11
3.4.1 CLASSIFICAÇÃO DOS CUSTOS	12
3.4.2 CUSTOS DO PRODUTO	12
3.4.3 PRECIFICAÇÃO	13
3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS	14
3.5.1 GERENCIANDO FINANÇAS	14
3.5.2 ESTUDANTES NA PRÁTICA	15
4. CONCLUSÃO	16
REFERÊNCIAS	17
ANEXOS	19

# 1. INTRODUÇÃO

O EasyPet é um aplicativo desktop projetado para simplificar a gestão de informações valiosas sobre animais de estimação. Este software é especialmente desenvolvido para interagir com coleiras inteligentes equipadas com sensores e armazenar dados cruciais em um banco de dados. Permitindo acesso rápido e fácil a informações detalhadas sobre a saúde, localização e atividades do animal de estimação, tudo apresentado de forma intuitiva em uma interface de usuário amigável.

Imagine ter a capacidade de rastrear o histórico médico do seu animal de estimação, incluindo vacinações, visitas ao veterinário e outros dados relevantes, tudo a partir de um único local.

Além disso, o EasyPet oferece recursos de rastreamento de localização em tempo real, permitindo saber exatamente onde o animal de estimação está a qualquer momento. Seja para monitorar seus passeios diários, identificar padrões de comportamento ou, em caso de emergência, localizar seu animal instantaneamente, proporcionando uma tranquilidade para os proprietários de animais de estimação.

Navegar pelo aplicativo é fácil e intuitivo, o EasyPet coloca as informações do animal de estimação ao seu alcance, permitindo tomar decisões informadas sobre sua saúde e bem-estar.

## **2. DESCRIÇÃO DA EMPRESA**

A Instituição de Ensino UNIFEOB, tem seu endereço em São João da Boa Vista, bairro Jardim Nova São João, Av. Dr. Otávio da Silva Bastos, 2439, no Estado de São Paulo, e o CEP é 13874-149. A razão social da empresa é Fundação de Ensino Octávio Bastos e o seu número de registro é 59.764.555/0001-52.

A Unifeob é um centro universitário que visa mais do que entregar ao mercado bons profissionais, ela prepara os seus estudantes para a vida. Carregando a motivação de seu fundador ela desenvolve competências, habilidades e atitudes em cada atividade proposta, proporcionando, uma formação integral dos estudantes.

### **3. PROJETO INTEGRADO**

Nesta etapa do PI serão apresentados os conteúdos que cada unidade de estudo utilizará para realizar o projeto, assim como a forma que serão aplicados na empresa escolhida para a realização do projeto.

#### **3.1 PROGRAMAÇÃO ORIENTADA A OBJETO**

A programação orientada a objetos é uma forma de se programar que aproxima o mundo real ao código que está sendo feito. Teve seu início no final da década de 60 com a linguagem Simula-68, e ficou popularmente conhecida com as linguagens C++ e Java no início dos anos 70.

Seus conceitos foram iniciados pelo cientista da computação Alan Kay, que além de sua participação na POO, foi um dos criadores da linguagem de programação Smalltek. Anos mais tarde, em 2003, Alan veio a receber o prêmio TURING pela fundamentação da Programação Orientada a Objetos.

"A programação orientada a objetos é uma forma de organizar o código que torna o software mais fácil de entender, manter e reutilizar." - (BOOCH, 1991) e "Falar é fácil, me mostre o código." (TORVAL, 2001). Podemos ligar as duas frases à organização de um sistema. Criar um software não é apenas torná-lo funcional, fazê-lo dar resultado, mas também mantê-lo organizado para uma fácil manutenção, tanto para o próprio criador quanto para algum outro programador que poderá ser responsável pelo projeto mais a frente. A orientação a objetos ajuda nesta parte, organização e facilidade de manutenção do código fonte.

##### **3.1.1 CLASSES E OBJETOS**

Classe e objetos são conceitos fundamentais na programação orientada a objetos (POO). Uma classe é uma estrutura que atua como um modelo para criar objetos, ela define um tipo de dado com atributos (variáveis) e métodos (funções) que operam nesses atributos.

Um objeto é uma instância de uma classe, ele é uma entidade que possui atributos específicos definidos pela classe e pode executar métodos especificados pela classe, cada

objeto criado a partir de uma classe é independente dos outros objetos e pode ter valores diferentes para seus atributos.

Conforme Cooper, "Classes e objetos fornecem uma maneira de modelar o mundo real em código, tornando o software mais próximo da realidade." Cooper quis dizer que as classes e objetos em programação são uma maneira de representar o mundo real no código. Isso torna o software mais fácil de entender e usar, pois os usuários podem se relacionar com os conceitos do mundo real.

### **3.1.2 ATRIBUTOS, MÉTODOS, ENCAPSULAMENTO, HERANÇA E POLIMORFISMO.**

A POO é um paradigma de programação que considera o software como um conjunto de objetos que interagem entre si. Os objetos são entidades que possuem atributos e métodos, que representam características e comportamentos do mundo real.

"Os atributos são características ou propriedades de um objeto. Eles são usados para armazenar dados sobre o objeto." (BOOCH; RUMBAUGH; JACOBSON, 2005, p. 22). Os atributos são uma parte importante da programação orientada a objetos (POO). Eles permitem que os desenvolvedores criem objetos que são mais fáceis de entender, usar e manter.

"Métodos são ações que um objeto pode realizar. Eles são usados para manipular os dados armazenados nos atributos." (GAMMA; HELM; JOHNSON; VLISSIDES, 1995, p. 23) explica que os métodos são rotinas que são usadas para realizar ações em um objeto. Eles são usados para manipular os dados armazenados nos atributos do objeto.

"O encapsulamento é um mecanismo que garante que os dados e comportamentos de um objeto sejam escondidos de outros objetos." (GAMMA; HELM; JOHNSON; VLISSIDES, 1995, p. 26) explica que o encapsulamento é um princípio de programação orientada a objetos (POO) que permite que os dados e comportamentos de um objeto sejam mantidos privados. Isso torna os objetos mais fáceis de entender, usar e manter.

"O código é herança. Devemos escrever código que seja fácil de entender e manter, para que as gerações futuras possam usá-lo e aprimorá-lo." (ROBERT C. MARTIN autor do livro "Clean Code: A Handbook of Agile Software Craftsmanship"). A herança de programação é uma ferramenta poderosa que pode ser usada para tornar o desenvolvimento de software mais eficiente e eficaz. Ao escrever código que seja fácil de entender e manter, os programadores podem deixar um legado duradouro para as gerações futuras.

"O polimorfismo é uma ferramenta poderosa que pode ser usada para criar código mais eficiente e eficaz. No entanto, é importante usá-lo com cuidado, pois pode levar a código



confuso e difícil de manter." (ECKEL, 2006, p. 44). Esta citação é uma boa representação do potencial e dos perigos do polimorfismo. Ela enfatiza que o polimorfismo pode ser uma ferramenta poderosa, mas que é importante usá-lo com cuidado.

### **3.1.3 MÉTODOS ESTÁTICOS, PÚBLICOS E PRIVADOS**

Conforme mencionado anteriormente, a orientação a objetos possui seus métodos e atributos, e através dele podemos utilizar do encapsulamento para tornar nossa aplicação mais produtiva e segura.

Falando dos modificadores de acesso estáticos, são aqueles que pertencem à classe, e não a uma instância. Isso significa que eles podem ser chamados sem criar uma instância da classe. Métodos estáticos são frequentemente usados para implementar funções que não dependem de nenhum estado da instância, como funções matemáticas ou de conversão. Por exemplo, falando sobre conexões com APIs no frontend ou solicitações ao banco de dados no backend, criar uma classe controladora de requisições não tem a necessidade de ser instanciada, pois esta não necessita de atributos.

Sobre os públicos, eles necessitam de uma instância e podem ser acessados de qualquer lugar fora da classe, e pode-se dizer que é um facilitador na hora de programar, porém pode se tornar um vilão quando se fala de produtividade, preocupações com erros e integridade. Há dados em uma classe que não pode ser acessada de fora de seu escopo, e que pode comprometer a segurança do código. Segundo Martin Fowler, em seu livro "Refatoração: Aperfeiçoando o Projeto de Código Existente", "Atributos públicos são como janelas de uma casa: eles permitem que você veja o que está dentro, mas também permitem que outras pessoas mudem o que está dentro."

Os privados são comumente usados na programação hoje em dia, e são métodos ou atributos que só podem ser acessados de dentro da classe em que são declarados. Eles são usados para implementar lógica que não deve ser acessível a código externo.

Em nosso código, utilizaremos bastante dos atributos privados em classes que não podem ter alteração direta, como por exemplo a classe Animal, que conforme utilizaremos não terá a necessidade de alterar seu nome fora requisições de atualização ao banco de dados.

## **3.2 LÓGICA DE PROGRAMAÇÃO**

A palavra lógica deriva do grego *lógos*, que significa "palavra", "discurso" ou "raciocínio". Essa palavra foi usada pelos filósofos gregos para se referir à razão e ao pensamento racional.

Na programação, lógica é o conjunto de regras e conceitos que definem como os programas de um computador devem ser estruturados e desenvolvidos. É a base para a criação de programas que sejam corretos, eficientes e fáceis de manter. Um programa correto é aquele que faz o esperado dele, já o eficiente usa os recursos do computador de forma eficaz.

Foi fundamental ao projeto principalmente na criação do banco de dados visto que, um deslize pode ser o suficiente para ter de reformulá-lo. Tal como, errar a relação de uma tabela com outra, o tipo de dado de um atributo ou esquecer de adicionar o auto-incremento quando preciso.

### **3.2.1 CONCEITOS FUNDAMENTAIS DO DESENVOLVIMENTO DE SOFTWARE**

Cada linguagem de programação tem suas diferenças, como sintaxe, semântica, aplicação, entre outros. Entretanto, a lógica utilizada em todas as linguagens se mantém a mesma, fazendo uso de uma sequência organizada de etapas para atingir um resultado esperado, também conhecido como algoritmo.

É importante ressaltar a diferença entre variáveis e constantes, a primeira é um espaço utilizado para armazenar algum tipo de dado que pode ser modificado, enquanto a segunda possui uma informação definida não podendo ser alterada. Existem diferentes tipos de dados, os principais sendo, int (números inteiros), float (números decimais), string (caracteres) e boolean (verdadeiro ou falso). Por exemplo, uma frequência de batimento cardíaco está em frequente alteração, portanto é uma variável e seu tipo é float.

Em lógica da programação, função é um bloco de código que pode ser reutilizado em diferentes partes do programa. Já estruturas condicionais são blocos de códigos que executam diferentes comandos dependendo do valor e da condição. Existem também os operadores, que são símbolos utilizados para realizar operações matemáticas, lógicas ou de comparação. Tal como, a verificação de usuário, nada mais é do que uma estrutura condicional onde há um operador relacional de igualdade, se as credenciais apresentadas forem iguais as registradas o usuário conseguirá usar o aplicativo, caso contrário não.

A principal diferença entre função e estrutura condicional é que a função pode ser reutilizada em diferentes partes do programa, enquanto a estrutura condicional é um bloco de código que executa diferentes comandos.

### **3.2.2 DESENVOLVIMENTO DE APLICAÇÕES DESKTOP**

Para o desenvolvimento das interfaces foram utilizadas as linguagens HTML e CSS, com o objetivo de apresentar os dados de maneira simples e eficaz, tendo como foco transmitir uma boa experiência ao usuário. Utilizando o MySQL no aplicativo é possível criar e gerenciar um banco de dados, nele estão todos os dados da coleira e do animal de estimação. Acrescentando a linguagem de programação JavaScript é possível conectar o banco de dados com as interfaces.

Como o intuito do aplicativo é passar as informações do arduíno para uma tela fez-se uso do CRUD, que é um acrônimo para Create, Read, Update, Delete, sendo essas as quatro operações básicas realizadas em um banco de dados. Por exemplo, usa-se o CRUD para criar o registro de um animal de estimação, ler e atualizar sua ficha médica e excluir seu registro caso necessário.

## **3.3 MODELAGEM DE DADOS**

A modelagem de dados é um processo essencial no desenvolvimento de sistemas de informação, pois é responsável por determinar os dados necessários para o funcionamento do sistema. De acordo com Martin (2009, p. 25), "a modelagem de dados é o processo de definir os dados necessários para apoiar um sistema de informação". Ela desempenha um papel crucial na definição de como os dados são armazenados, organizados, acessados e relacionados dentro de um sistema de informação.

### **3.3.1 MODELO CONCEITUAL**

Um modelo conceitual é a primeira parte no desenvolvimento da modelagem de dados, ele é usado para garantir que os requisitos de negócios sejam compreendidos e documentados de forma precisa, é também usado como base para modelos lógicos e físicos. Dentro do projeto foi utilizado com o intuito de criar uma base para os modelos lógico e físico, sua principal função foi a organização de dados, conforme mostrado no Anexo A.

### **3.3.2 MODELO LÓGICO E FÍSICO**

O modelo lógico é uma tabela que mostra a estrutura do banco de dados, enquanto o físico acaba sendo a reprodução de um sistema real.

O modelo lógico é uma representação abstrata dos dados, independente de qualquer implementação específica. O modelo físico é uma representação concreta dos dados, que considera as restrições do sistema de gerenciamento de banco de dados. Silberschatz, Korth e Sudarshan (2011, p. 43)

A principal função do modelo lógico foi possibilitar a implementação de entidades com seus atributos na tabela, além da conexão entre elas, também chamada de relacionamento. Enquanto o modelo físico permitiu a implementação do lógico ao banco de dados, além de garantir maior integridade dos dados.

Empregando o software MySQL tornou-se capaz a criação e gerenciamento dos modelos, lógico e físico, do projeto. Em ambos, os dados preenchidos constam os registros do arduino, informações sobre o animal que fará uso da coleira, o cadastro do usuário e as datas caso ocorra uma locação da coleira.

### **3.3.3 SQL**

De acordo com Nield (2022, p. 3), "SQL é uma linguagem poderosa e flexível que pode ser usada para uma variedade de tarefas, desde a criação de relatórios simples até a análise de dados complexos.". Teve seu uso no projeto para popularização do banco de dados, mais especificamente utilizando o CRUD (Create Read Update Delete) , preenchemos os dados e definimos a maneira a serem apresentadas as tabelas.

## **3.4 GESTÃO FINANCEIRA**

Gestão Financeira pode ser definida como a gestão dos fluxos monetários derivados da atividade operacional, em termos de suas respectivas ocorrências no tempo. (CHENG; MENDES, 1989). Ela é fundamental independente da organização. São envolvidos planejamentos e controle de recursos de um projeto ou empresa. Uma parte indispensável é a análise de projetos e a avaliação da viabilidade de se colocar em produção. Através do conhecimento em gestão financeira pode-se determinar o futuro de um produto.

Se uma empresa não planejar a qualidade de produtos e serviços que oferece, optou por não oferecê-la. Uma empresa sem um bom planejamento financeiro não tem a gestão do seu próprio comércio. A partir do gerenciamento de gastos, pode-se tomar melhores decisões sem riscos à companhia. Vale ressaltar a perspectiva de futuro e tomadas de decisões.

Neste sentido, a função financeira na indústria e no comércio está associada ao apoio e controle das demais funções da empresa e à viabilização financeira das estratégias empresariais.

### **3.4.1 CLASSIFICAÇÃO DOS CUSTOS**

No que se diz respeito a custos, temos o fixo e variável. O custo fixo pode ser definido como aquele que não varia independente da atividade da empresa, ou seja, permanecerá o mesmo que o volume de vendas mude (LEONE, 2000, p.55). Já o variável, têm suas alterações conforme o ritmo do mercado. Tendem a oscilar seu crescimento em proporção às mudanças no nível de atividade. (BACKER ; JACOBSEN, 1974, p.13)

Ele é subdividido em outros tipos de custos, como o direto e indireto. O custo direto se refere a todo valor destinado à produção de um produto específico, como por exemplo a mini câmera, os fios de um arduíno ou o acelerômetro.

Custos diretos são os custos que podem ser identificados e quantificados no produto ou serviço e valorizados com relativa facilidade. Os materiais diretos, por exemplo, são normalmente requisitados com a identificação prévia de sua utilização, ou seja, ao emitir a requisição para o almoxarifado, o responsável pela produção já que não podem ser identificados de forma fácil, não podem ser apropriados de forma direta para as unidades específicas como, por exemplo, mão-de-obra indireta e matérias indiretas. Schier (2006, p.26)

Já o indireto, é tudo relacionado exclusivamente à produção, e que não se refere a um produto direto, por exemplo, o salário do supervisor, energia elétrica e água sendo todos do setor de produção.

Custos indiretos são aqueles necessários à produção, mas alocáveis arbitrariamente, através de um sistema de rateio, estimativas e outros meios. (MARION; SEGATTI, 2009, p. 43).

### **3.4.2 CUSTOS DO PRODUTO**

No projeto, identificamos custos diretos e indiretos para realizar o cálculo do custo total de nosso produto.

O custo direto se aproxima dos R\$4.500,00 mensais (30 coleiras), já incluso a manutenção e hospedagem do software, sendo o unitário do equipamento R\$144,26 e um custo médio do servidor em R\$135,00 mensais (AWS).

#### **Tabela 1 - Custos**

MOTIVO	VALOR	LINK	TIPO CUSTO
FIOS	R\$ 14,90	<a href="#">Acessar</a>	CD
ACELEROMETRO	R\$ 18,50	<a href="#">Acessar</a>	CD
MINI CAMERA	R\$ 25,28	<a href="#">Acessar</a>	CD
ARDUINO	R\$ 15,29	<a href="#">Acessar</a>	CD
SENSOR BATIMENTOS CARDIACOS	R\$ 8,90	<a href="#">Acessar</a>	CD
SENSOR TEMPERATURA CORPORAL	R\$ 13,90	<a href="#">Acessar</a>	CD
COLEIRA PEITORAL	R\$ 29,90	<a href="#">Acessar</a>	CD
CHIP LOCALIZAÇÃO	R\$ 17,59	<a href="#">Acessar</a>	CD
CUSTO SOFTWARE	R\$ 135,00		CI
MAO DE OBRA	R\$ 6.000,00		CI
AGUA, LUZ, TELEFONE, INTERNET	R\$ 250,00		CI
DOMINIO	R\$ 5,00		CI

Fonte: Autoral.

Como mostra a tabela acima, procuramos as matérias-primas que melhor se qualificam para nosso produto, tanto em questão de qualidade, quanto de preço, pois, nossa prioridade atualmente é ganhar espaço no mercado de vendas e para isso precisamos ganhar a confiança de nossos clientes.

Realizamos uma pesquisa de mercado para verificar os concorrentes e analisar possíveis custos que teriam. É fundamental compreender a estrutura de custos dos concorrentes, a semelhança entre os produtos oferecidos e o número e tamanho das outras lojas. (COELHO, 2012).

O custo unitário de nosso produto foi de R\$457,26, e pretendemos vender por R\$600,00 e, Por meio da pesquisa realizada, verificamos a viabilidade do nosso produto referente a todo custo-benefício que ele fornece aos nossos clientes, e para isso, usamos como exemplo a marca INVOXIA, que vende um produto similar pelo valor aproximado de R\$800,00.

### 3.4.3 PRECIFICAÇÃO

Segundo Churchill Jr. e Peter (2013), o preço é definido como a quantia de dinheiro necessária para adquirir a propriedade ou o direito de consumir ou utilizar um produto ou serviço.

A precificação é uma das partes mais importantes na hora de se idealizar e colocar um produto em prática, através dela descobrimos a margem de contribuição, sendo definida por Cogan (1999, p.24) como “A diferença, para mais, entre o preço de venda e as despesas variáveis referentes às unidades vendidas”, e o markup que, segundo Mazeto (2018), é um índice usado para a formação do preço de venda, aplicado sobre o custo dos produtos, sendo necessário incluir as despesas, os impostos e os demais encargos que incidem na venda.

**Tabela 2 - Formação de Preço**

Produto / Serviço	Custo Fixo	Custo Variável		Custo Total	Preço	Margem de Contribuição	Margem	Mark Up	Índice de Markup
		Matéria Prima	Impostos						
EasyPET	R\$ 253,00	R\$ 144,26	R\$ 60,00	R\$ 457,26	R\$ 600,00	R\$142,74	23,79%	31,22%	1,31

Fonte: Autoral

Nos dados acima, realizamos as contas da precificação. O custo fixo é de R\$7590,00, porém dividimos este valor pela quantidade de vendas ( 30 unidades iniciais ) para chegarmos no valor por unidade e utilizamos 10% destinado a impostos sobre as vendas.

No cálculo final, a margem de lucro seria de 23,79% com o markup de 31,22%, garantindo uma saúde estável para o financeiro da empresa.

### 3.5 CONTEÚDO DA FORMAÇÃO PARA A VIDA: GERENCIANDO FINANÇAS

A Formação para a Vida é um dos eixos do Projeto Pedagógico de Formação por Competências da UNIFEQB.

Esta parte do projeto está diretamente relacionada com a extensão universitária, ou seja, o objetivo é que seja aplicável e que tenha real utilidade para a sociedade, de um modo geral.

#### 3.5.1 GERENCIANDO FINANÇAS

A compreensão dos conceitos fundamentais da economia e finanças é crucial para uma gestão financeira eficaz. A economia, que estuda a produção, distribuição e consumo de bens e serviços, é intrinsecamente ligada às finanças, o ramo que se dedica à administração do dinheiro. Ativos, representando tudo com valor econômico, contrastam com passivos, que englobam obrigações como dívidas. O patrimônio líquido, diferença entre ativos e passivos, é um indicador crucial. Receitas são entradas de dinheiro, enquanto despesas representam saídas. Investimentos, aplicação de dinheiro para obter lucro, envolvem riscos e buscam retornos benéficos.

A busca pela independência financeira, habilidade de sustentar-se sem depender de emprego, está atrelada ao valor da riqueza, não apenas o quantitativo de bens, mas também a capacidade de gerar renda. Imagine o seguinte exemplo: ao investir em uma criptomoeda,

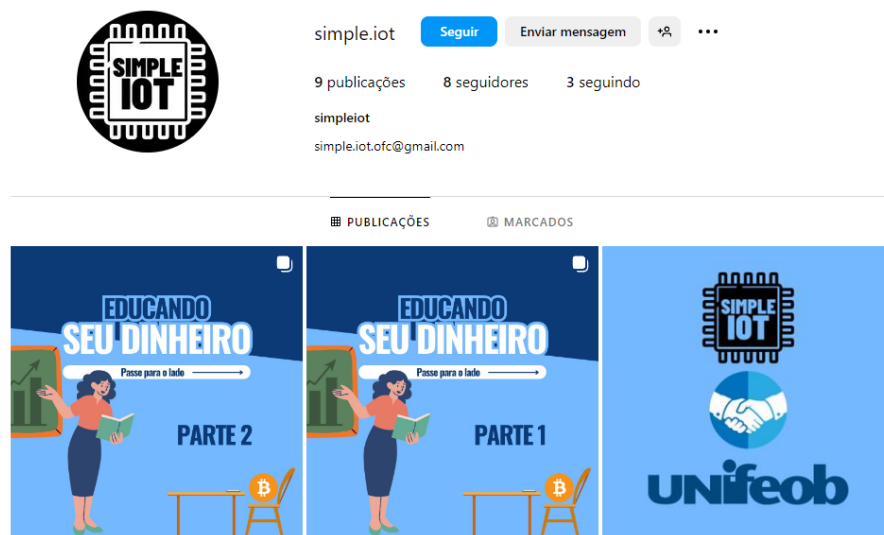
you run the risk of losing the investment, but by making a correct analysis it becomes easier to get profit.

Debts, loans with interest, and complex compound interest require attention. Loan options, such as financing, special check and credit card, present alternatives to indebtedness. Consider the following example: by using the credit card responsibly and paying the full balance each month, you avoid the high interest rates that normally accompany this form of credit. Strategies such as renegotiation, consolidation and portability of the debt offer ways to deal with financial challenges.

Defining goals is the starting point for achieving financial objectives, being crucial to establish deadlines and create concrete plans. Imagine that your goal is to buy a house. Define specific goals, such as saving a certain amount monthly, set realistic deadlines and create a detailed plan to reach this goal.

### 3.5.2 ESTUDANTES NA PRÁTICA

For the project, we made a banner explaining briefly about financial management and some tips on how to use your money correctly. We created three posts and put them in our business Instagram. See the image:



It was very interesting content and even more the opportunity to pass this knowledge on.



## 4. CONCLUSÃO

O objetivo do projeto é criar um aplicativo desktop que pegue as informações da coleira, no banco de dados, e mostre para o usuário em uma tela. Será possível criar uma conta com login e senha para o cliente, o registro do animal de estimação e sua ficha de atendimento, além de poder acessar e alterar os dados se necessário.

Contudo, apenas fazer um aplicativo não basta, torna-se necessário analisar os custos gerais, permitindo ter uma noção do valor final e sua viabilidade no mercado. Fazendo uso da gestão financeira, conclui-se que o preço de venda é menor que o da marca concorrente, resultando em maior acessibilidade por parte dos clientes, possibilitando crescer no mercado.

## REFERÊNCIAS

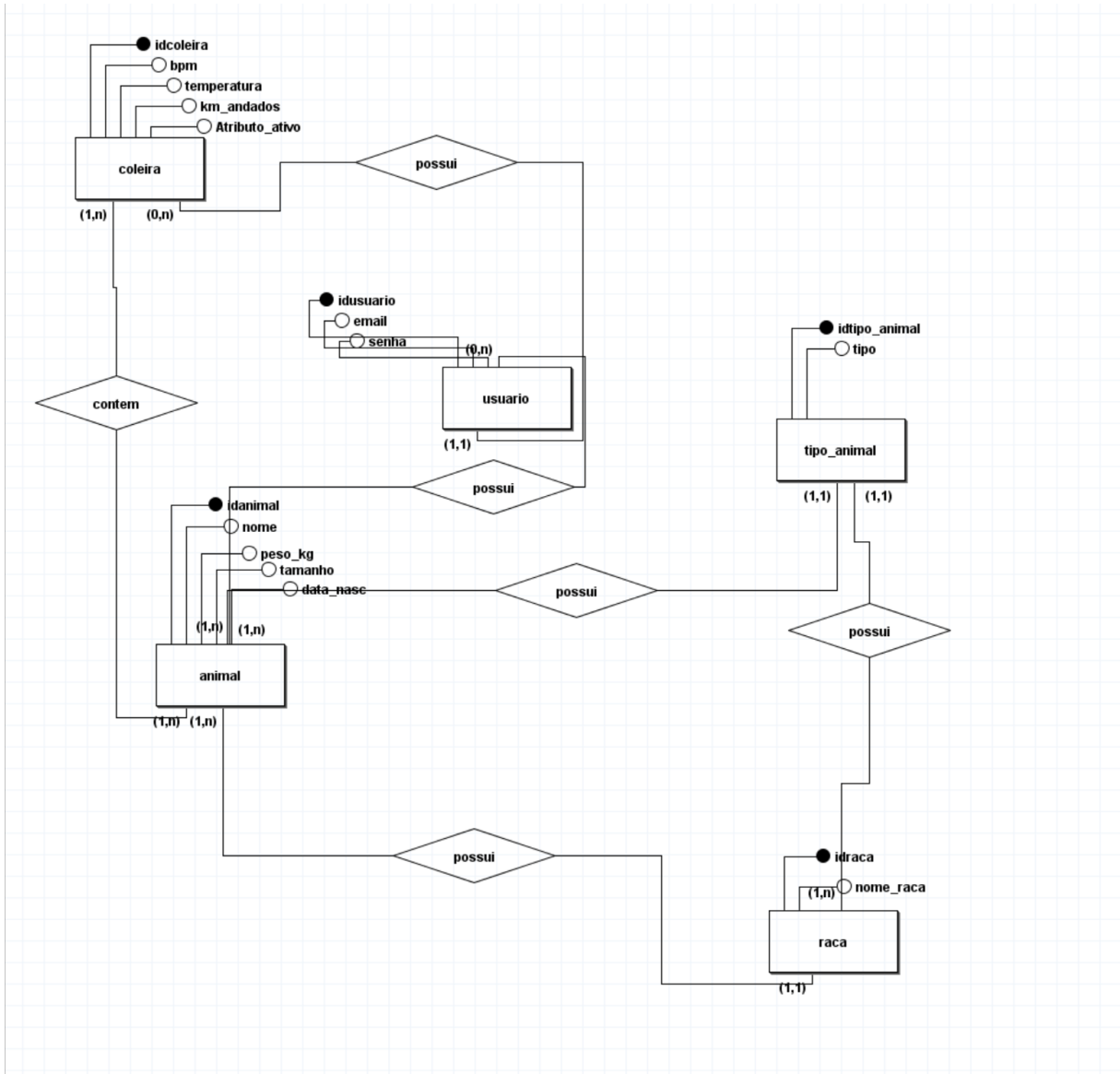
- BACKER, Morton; JACOBSEN, Lyle E. **Contabilidade de Custos: Um Enfoque de Administração de Empresas**. Volumes 1 e 2. Tradução de Pierre Louis Laporte. São Paulo: McGraw-Hill do Brasil, 1974. Acesso em 19 out. 2023.
- BOOCH, GRADY. "**The Object-Oriented Paradigm.**" Palestra apresentada na ACM SIGSOFT Symposium on Software Engineering, 1991.
- CHENG A., MENDES M. **A importância e responsabilidade da gestão financeira na empresa**. 1989. Disponível em: <<https://www.scielo.br/j/cest/a/K537QpqPKNmpTf4CVsh5CPc/#>>. Acesso em: 05 out. 2023.
- CHURCHILL Jr., G. A.; PETER, J. P. **Marketing: criando valor para o cliente**. 3. ed. São Paulo: Saraiva, 2013. Acesso em: 21/09/2023.
- COGAN, S. **Custos e preços: formação e análise**. São Paulo. Pioneira, 1999. Acesso em: 19/10/2023
- FOWLER, Martin. **Refatoração: Aperfeiçoando o Projeto de Código Existente**, p. 29. 1999. Acesso em: 23/10/2023.
- LEONE, George Sebastião Guerra. **Custos: Planejamento, Implantação e Controle**. São Paulo: Atlas, 2000. Acesso em 19/10/2023.
- MARTIN, James. **Sistemas de informação: uma abordagem gerencial**. 9. ed. São Paulo: Pearson Prentice Hall, 2009.
- MAZETO, Thiago. **Markup: entenda e calcule corretamente**. Disponível em: <<https://www.ecommercebrasil.com.br/artigos/markup-entenda-e-calcule-corretamente>>, 2018. Acesso em 19/10/2023.
- NIELD, Thomas. **Introdução à Linguagem SQL: Abordagem prática para iniciantes**. 2. ed. São Paulo: Novatec, 2022. p. 3.
- SCHIER, Carlos Ubiratan da Costa. **Gestão de custos**. Curitiba: Ibpex, 2006. Acesso em 19/10/2023
- SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Fundamentos de sistemas de banco de dados**. 6. ed. São Paulo: Pearson Prentice Hall, 2011.
- SOARES, Thiago Coelho. **Finanças, Custos e Formação do Preço de Venda**. 2012. Acesso em: 05/10/2023.
- TORVALDS, LINUS. "**Falar é fácil, me mostre o código.**" Postagem no fórum de discussão do kernel do Linux, 2001.

Booch, G., Rumbaugh, J., & Jacobson, I. (2005). **The Unified Modeling Language Reference Manual** (3rd ed.). Addison-Wesley.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). **Design Patterns: Elements of Reusable Object-Oriented Software**. Addison-Wesley.

## ANEXOS

## ANEXO A - MODELO LÓGICO BANCO DE DADOS



## ANEXO B - MODELO CONCEITUAL BANCO DE DADOS

